CC viva

Cloud Computing with Example: Cloud computing is the delivery of computing services over the internet. For example, Google Drive allows users to store files online and access them from any device with internet access.

Father of Cloud Computing: Dr. Larry Ellison, the founder of Oracle Corporation, is often credited as one of the pioneers of cloud computing.

Basic Characteristics of Cloud Computing: On-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service.

Vertical and Horizontal Scaling in Cloud Computing: Vertical scaling increases the capacity of a single server, while horizontal scaling adds more servers to distribute the load.

Services Provided by Cloud Computing: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

Deployment Models in Cloud Computing: Public cloud, private cloud, hybrid cloud, and community cloud.

Platforms for Large Scale Cloud Computing: Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), and IBM Cloud.

Large Cloud Providers and Databases: AWS (Amazon Web Services) provides services like Amazon RDS, Amazon Redshift, etc. Microsoft Azure offers Azure SQL Database, Cosmos DB, etc.

Difference between Cloud and Traditional Datacenters: Cloud computing offers scalability, flexibility, and cost-effectiveness compared to

traditional datacenters, which require upfront investment in hardware and maintenance.

Computing Components Required in Cloud: Virtualization, storage, networking, security, management tools, and applications.

Platforms of Cloud Architecture: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

Cloud Service: A cloud service is any service made available to users on demand via the internet from a cloud computing provider's servers.

Basic Clouds in Cloud Computing: Public cloud, private cloud, hybrid cloud, and community cloud.

Issues with Cloud Computing: Security concerns, data privacy, compliance challenges, and potential downtime.

Services Provided by Windows Azure Operating System: Windows Azure offers services like virtual machines, app services, storage, databases, and more.

Cloud Computing Architecture: Cloud computing architecture refers to the components and subcomponents required for cloud computing.

Services Provided by AWS: AWS offers a wide range of services including computing power, storage, databases, machine learning, and more.

Services Provided by Microsoft: Microsoft provides services like Azure Virtual Machines, Azure App Service, Azure SQL Database, etc.

Virtualization: Virtualization is the process of creating a virtual version of something, such as an operating system, server, storage device, or network resource.

EC2 Instance: An EC2 instance is a virtual server in Amazon's Elastic Compute Cloud (EC2) service.

AMI in EC2: An Amazon Machine Image (AMI) is a template used to create virtual servers (EC2 instances) in Amazon Web Services (AWS).

Launching an On-Demand EC2 Instance in AWS: Log in to the AWS Management Console, choose EC2, click on "Launch Instance," select an AMI, choose an instance type, configure instance details, add storage, configure security groups, review and launch the instance.

Google App Engine: Google App Engine is a Platform as a Service (PaaS) offering for

developing and hosting web applications on Google's infrastructure.

Advantages of Google App Engine: Scalability, built-in services like data storage, user authentication, and easy integration with other Google Cloud services.

Steps to Install and Configure Google App Engine: Install the Google Cloud SDK, create a project in the Google Cloud Console, configure your development environment, and deploy your application using the gcloud command-line tool.

Apex: Apex is a programming language used for developing applications on the Salesforce platform.

Features of Apex as a Language: Object-oriented, strongly typed, and supports transaction control and exception handling.

Applications of Apex: Apex is used for developing custom business logic, triggers, and automation on the Salesforce platform.

Apex Code Development Tools: Salesforce Developer Console, Salesforce Extensions for Visual Studio Code, and Force.com IDE.

Steps to Create Application Using Apex Programming Language: Write Apex code, test it in a sandbox environment, deploy it to production, and monitor its performance.

Salesforce.com Inc.: Salesforce.com Inc. is a cloud-based software company known for its customer relationship management (CRM) platform.

Lightning Platform: Salesforce Lightning Platform is a development platform that allows users to build custom applications and integrations on top of Salesforce.

Creating Custom Application Using Salesforce Classic: Navigate to Setup, click on Build, select Create, and choose "Apps" to create a custom application using Salesforce Classic.

Difference Between Custom Application and Console Application in Salesforce: Custom applications are user-defined applications tailored to specific business needs, while console applications provide a unified workspace for handling multiple records simultaneously.

Steps to Create Custom Application Using Salesforce: Define the application's objectives, design the user interface, configure data access and security settings, and deploy the application.

Hypervisor in Cloud Computing: A hypervisor is a software layer that allows multiple virtual machines to run on a single physical machine.

Load Balancing in Cloud Computing: Load balancing distributes incoming network traffic across multiple servers to ensure optimal resource utilization and prevent overload on any single server.

Open-Source Cloud Computing Platform Databases: Examples include PostgreSQL, MySQL, and MongoDB.

Recent Trends in Cloud Computing: Serverless computing, multi-cloud strategies, edge computing, and AI/ML integration.

Applications of Cloud Computing: Cloud computing finds applications in various fields such as data storage and backup, software development and testing, big data analytics, IoT, and more.

AI viva

DFS vs. BFS:

DFS (Depth-First Search) explores as far as possible along each branch before backtracking.

BFS (Breadth-First Search) explores all neighbor nodes at the present depth before moving to the nodes at the next depth level.

Time and Space Complexity:

DFS: Time - O(V + E) (where V is the number of vertices and E is the number of edges), Space - O(V)

BFS: Time - O(V + E), Space - O(V)

Data Structures:

DFS typically uses a stack (can be implemented using recursion as well).

BFS typically uses a queue.

Best First Search vs. A Algorithm*:

Best First Search selects the node with the lowest heuristic value.

A* Algorithm combines the benefits of both uniform cost search and greedy best-first search by using a heuristic to prioritize nodes.

Drawback of Non-Heuristic Method for 8 Puzzle:

Non-heuristic methods for solving the 8 Puzzle problem might not efficiently find the optimal solution or may take an impractical amount of time for large instances due to the search space explosion.

Time and Space Complexity of Selection Sort:

Time Complexity: O(n^2) (where n is the number of elements)

Space Complexity: O(1)

Maximum Number of Comparisons in One Iteration:

For an array of size N, the maximum number of comparisons in one iteration is N - 1.

Stable Sort Algorithm:

A stable sort algorithm preserves the relative order of equal elements. Selection sort is not stable.

Constraints for N Queen Problem:

The constraints involve placing N queens on an N×N chessboard such that no two queens threaten each other.

Comparison: Backtracking vs. Branch and Bound:

Backtracking explores all possible solutions, while Branch and Bound uses pruning to discard

partial solutions that cannot lead to a better solution.

Constraint Satisfaction Problem:

It involves finding a solution that satisfies a set of constraints that specify allowable combinations of values for the variables.

Use of a Chatbot:

Chatbots are used for automating tasks, providing customer support, answering queries, and engaging users in conversations.

Dialog Flow:

Dialog flow is the sequence of interactions between a user and a chatbot that leads to accomplishing a task or providing information.

Requirements for Developing a Chatbot:

Clear objectives, knowledge base, natural language processing capabilities, user interface,

integration with messaging platforms, and continuous improvement mechanisms.

Chatbot Performance Evaluation:

Metrics such as accuracy, response time, user satisfaction, and task completion rate are used to evaluate chatbot performance.

Improving Chatbot Accuracy:

Train the chatbot with more data, fine-tune the natural language processing models, handle ambiguous queries better, and incorporate user feedback for continuous improvement.